

Ändert der NetWorker die Zugriffszeit einer Datei ?

Dies ist einer der am häufigsten gestellten Fragen. Um sie zu beantworten, muß ich allerdings etwas ausholen.

Hintergrund

Ein UNIX/Linux Dateisystem weist jeder Datei immer 3 Zeitstempel zu:

atime	Access time Die Zeit des letzten Zugriffs ändert sich immer, wenn die Datei gelesen wird
ctime	Change time Diese Zeit ändert sich immer, wenn die Dateiattribute geändert wurde.
mtime	Modification time Diese Zeit ändert sich immer, wenn die Datei geändert wurde

Diese Information wird nicht in der Datei selbst, sondern in der Inode information gespeichert.

Die nächste Frage liegt auf der Hand: Sind *mtime* und *ctime* nicht das gleiche? - Das stimmt beinahe:

- Wird die Datei geändert (dies paßt *mtime* an), dann ändert sich auch *ctime*.
- Aber *ctime* ändert sich auch dann, wenn nur die Datei-Attribute geändert werden. Das ist zum Beispiel dann der Fall, wenn der Besitzer der Datei wechselt. Durch diesen Vorgang ändert sich der Dateinhalt selbst natürlich nicht.

Wie können Sie die Zeitstempel überprüfen ?

Wichtig ist, daß Sie die richtigen Werkzeuge verwenden - sie dürfen die Zeitstempel selbst nicht ändern. Der UNIX/Linux Befehl `ls` ist bereits sehr gut dafür geeignet:

<code>ls -l</code>	zeigt die mtime an
<code>ls -lc</code>	zeigt die ctime an
<code>ls -lu</code>	zeigt die atime an

Zur Veranschaulichung der verschiedenen Zeitstempel werde ich eine einfache Textdatei benutzen. Nachdem die Datei erstellt wurde, sind natürlich alle Zeitstempel gleich:

```
bash-2.03# ls -l
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lc
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
```

Nach ein paar Minuten wird die Datei gelesen:

```
bash-2.03# more file

This is a test.

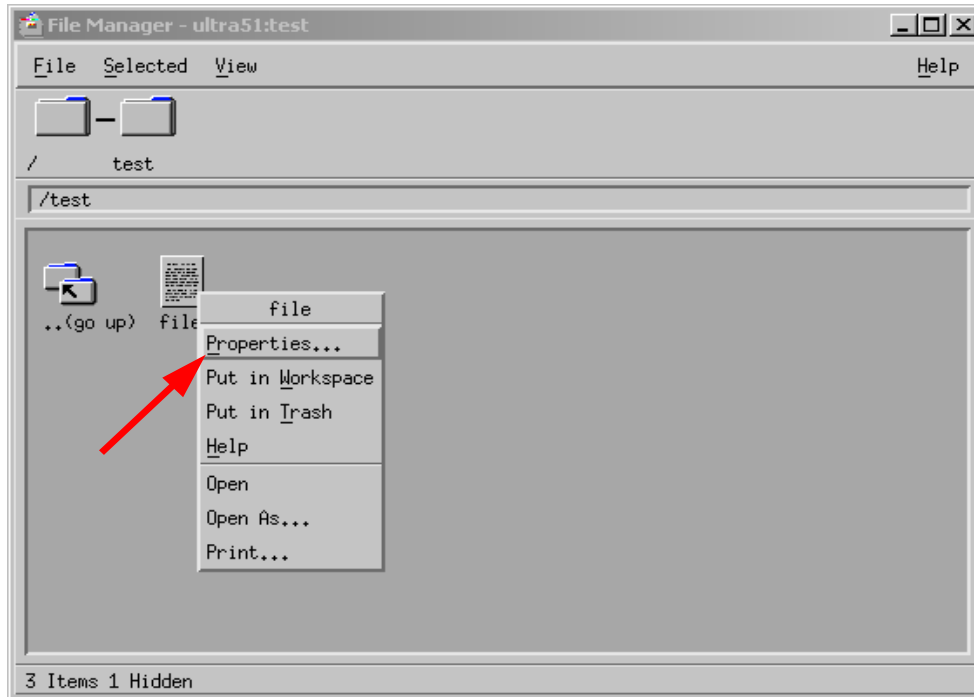
bash-2.03#
bash-2.03# ls -l
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lc
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:39 file ← .....
bash-2.03#
```

Wie sie sehen, hat sich hierdurch die *atime* verändert.

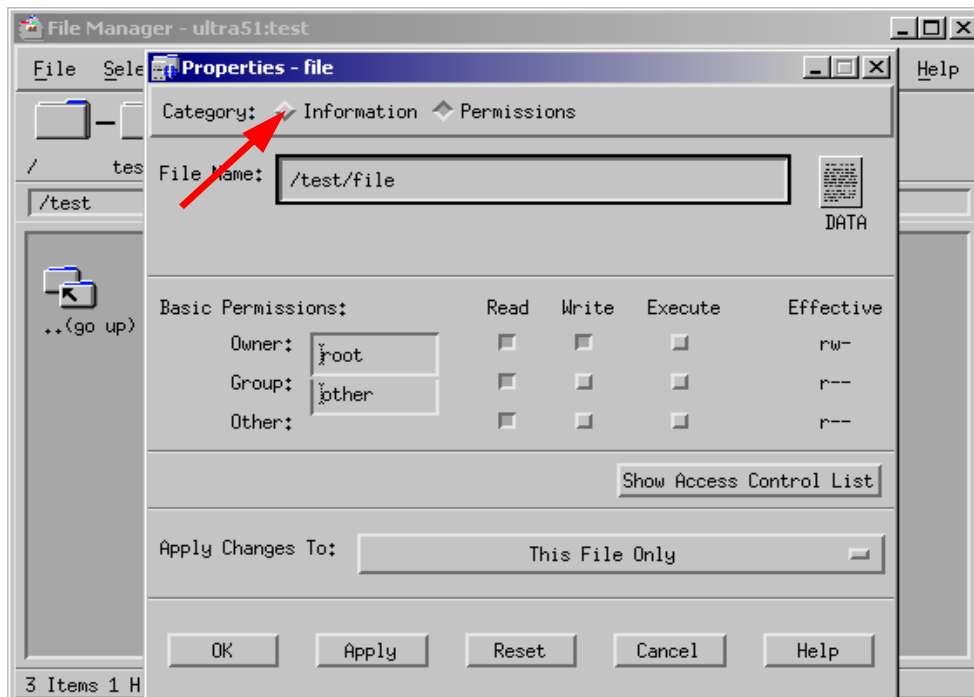
Wenn Sie 'nur' ein Attribut ändern, beeinflusst das lediglich die *ctime*:

```
bash-2.03# chmod +w file
bash-2.03#
bash-2.03# ls -l
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lc
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:51 file ← .....
bash-2.03#
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root    other      18 Mar  1 21:39 file
bash-2.03#
```

Wie ich bereits erwähnte, müssen Sie bei der Wahl Ihrer Werkzeuge vorsichtig sein. Wenn Sie zum Beispiel die Zeiten mit dem Befehl **File Manager - Properties...**

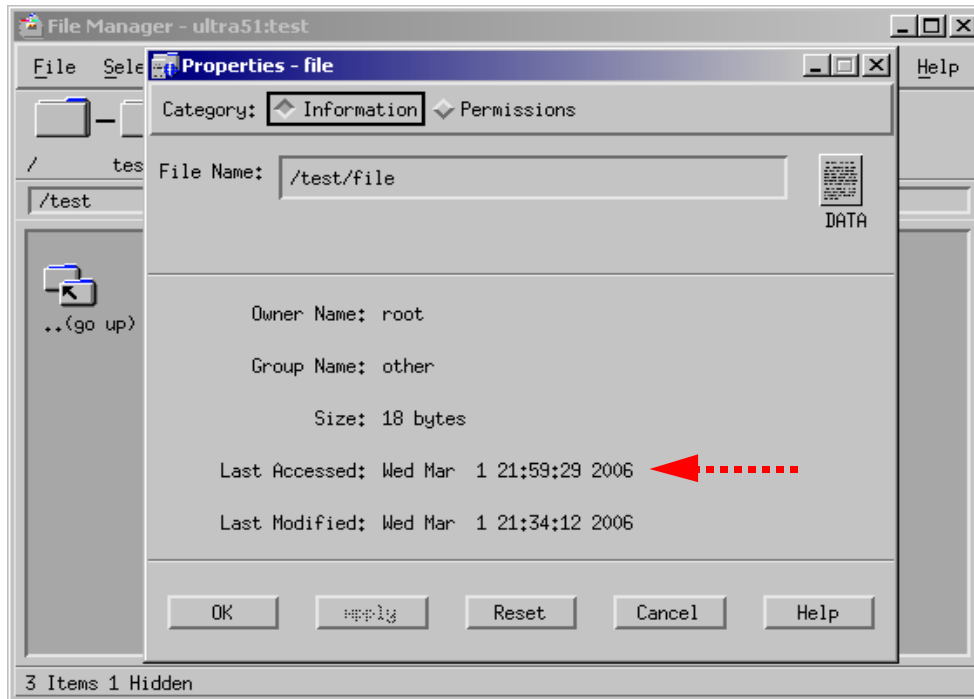


... und dort mit der Option **Information** überprüfen ...

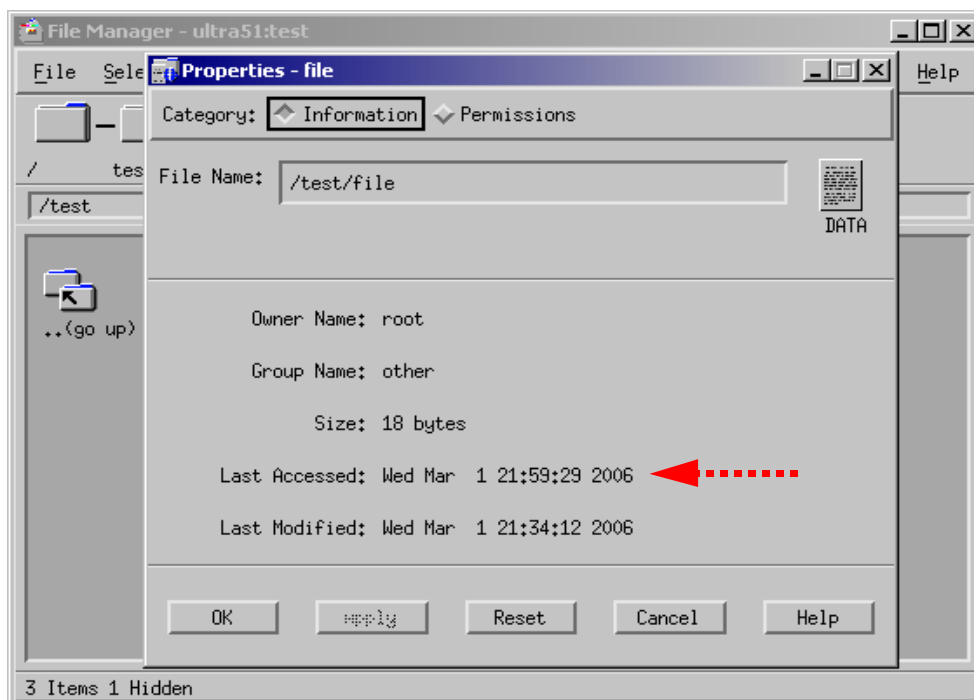


... können Sie dann *atime* und *mtime* feststellen (siehe nächste Seite).

Beachten Sie bitte, daß sich *atime* hierdurch geändert hat, obwohl Sie die Datei nicht gelesen haben:



Wiederholen Sie den Vorgang nach einigen Minuten, ändert sich *atime* erneut:



Und hier ist der Beweis:

```
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root    other      18 Mar  6 22:08 file
bash-2.03#
```

Mit diesem Wissen können Sie jetzt testen, ob und wie der NetWorker die Zeiten beeinflusst.
Fragen Sie erst die aktuellen Parameter ab:

```
bash-2.03# ls -l
total 2
-rw-r--r--  1 root      other          18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lc
total 2
-rw-r--r--  1 root      other          18 Mar  1 21:51 file
bash-2.03#
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root      other          18 Mar  1 22:08 file
bash-2.03#
```

Sichern Sie jetzt die Datei ...

```
bash-2.03# save /test
save: Using ultra51 as server
/test/file
/test/
/

save: /test  2 KB 00:00:03      3 files
bash-2.03#
```

... und wiederholen Sie die Anfrage nach Zeitstempeln:

```
bash-2.03# ls -l
total 2
-rw-r--r--  1 root      other          18 Mar  1 21:34 file
bash-2.03#
bash-2.03# ls -lc
total 2
-rw-r--r--  1 root      other          18 Mar  1 21:51 file
bash-2.03#
bash-2.03# ls -lu
total 2
-rw-r--r--  1 root      other          18 Mar  1 22:08 file
bash-2.03#
bash-2.03# date
Mon Mar  1 22:43:05 CET 2006
bash-2.03#
```

Wie Sie sehen, ändert der NetWorker die *atime* nicht.

Und wie verhält es sich unter Windows ?

Windows kennt die Abkürzungen aus der UNIX-Welt nicht und auch keine *change time*. Stattdessen gibt es unter Windows diese Zeitstempel:

Creation time

Diese Zeit wird gesetzt, wenn die Datei erstellt wurde.

Modification time

Diese Zeit wird gesetzt, wann immer die Datei geändert wurde.

Access time

Diese Zeit hält fest, wann die Datei zuletzt gelesen wurde.

Die einfachste Überprüfung der Zeitstempel geschieht unter Windows mit dem Pendant zum UNIX Befehl `ls`, also mit dem Befehl `dir`:

<code>dir</code>	berichtet die Modification time
<code>dir /ta</code>	berichtet die Access time
<code>dir /tc</code>	berichtet die Creation time
<code>dir /tw</code>	berichtet die Modification time

Die folgenden Seiten zeigen entsprechende Beispiele.

Nach der Erstellung einer Date sind wieder alle Zeiten gleich:

```
D:\test>copy con: file.txt
```

```
This is a test.
```

```
^Z
```

```
1 file(s) copied.
```

```
D:\test>
```

```
D:\test>dir /ta file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
              1 File(s)                21 bytes
              0 Dir(s)  2,606,137,344 bytes free
```

```
D:\test>
```

```
D:\test>dir /tc file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
              1 File(s)                21 bytes
              0 Dir(s)  2,606,137,344 bytes free
```

```
D:\test>
```

```
D:\test>dir /tw file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
              1 File(s)                21 bytes
              0 Dir(s)  2,606,137,344 bytes free
```

```
D:\test>
```

Das Lesen der Datei soll die Zugriffszeit ändern:

```
D:\test>type file.txt
```

```
This is a test.
```

```
D:\test>
```

```
D:\test>dir /ta file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)  2,606,129,152 bytes free
```

```
D:\test>
```

```
D:\test>dir /tc file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)  2,606,129,152 bytes free
```

```
D:\test>
```

```
D:\test>dir /tw file.txt
```

```
Volume in drive D is DATA
```

```
Volume Serial Number is 4851-E9F5
```

```
Directory of D:\test
```

```
03/01/2006  11:07 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)  2,606,129,152 bytes free
```

```
D:\test>
```

Aber wie Sie sehen, ändert sich gar nichts.

Um die Zeit zu ändern, soll jetzt ein Datei-Attribut geändert werden:

```

D:\test>attrib file.txt
A             D:\test\file.txt

D:\test>
D:\test>attrib +R file.txt

D:\test>
D:\test>dir /ta file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

03/01/2006  11:11 PM                21 file.txt  ◀-----
              1 File(s)                  21 bytes
              0 Dir(s)  2,605,993,984 bytes free

D:\test>
D:\test>dir /tc file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

03/01/2006  11:07 PM                21 file.txt
              1 File(s)                  21 bytes
              0 Dir(s)  2,605,993,984 bytes free

D:\test>
D:\test>dir /tw file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

03/01/2006  11:07 PM                21 file.txt
              1 File(s)                  21 bytes
              0 Dir(s)  2,605,993,984 bytes free

D:\test>

```

Und hierdurch wird tatsächlich die *Access time* geändert

Lassen Sie uns wieder überprüfen, ob eine NetWorker Sicherung einen der Zeitstempel ändert:

```
D:\test>save D:\test
save: Using de-creinfeld as server
D:\test\file.txt
D:\test\
D:\
/

save: D:\test  3 KB 00:00:32      4 files
save completion time:  3-06-06 11:36p

D:\test>
D:\test>dir /ta file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

03/01/2006  11:11 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)      2,605,862,912 bytes free

D:\test>
D:\test>dir /tc file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

03/01/2006  11:07 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)      2,605,862,912 bytes free

D:\test>
D:\test>dir /tw file.txt
Volume in drive D is DATA
Volume Serial Number is 4851-E9F5

Directory of D:\test

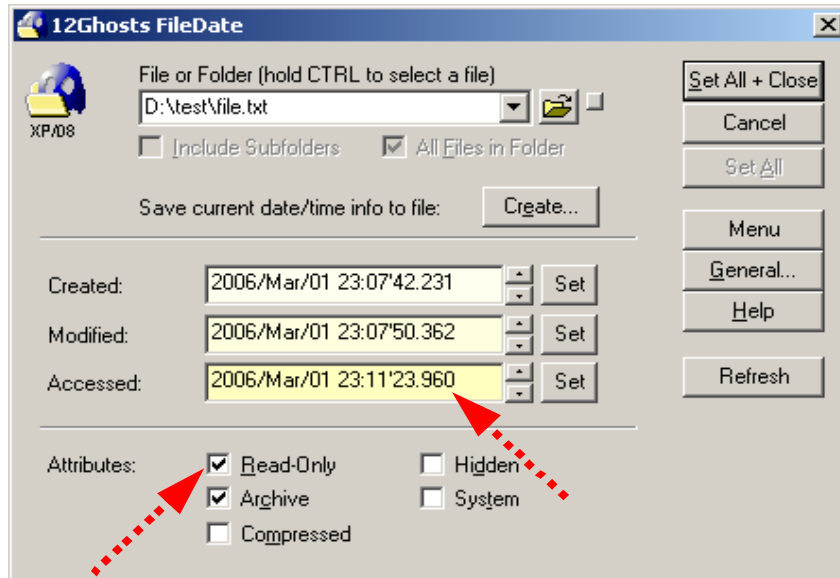
03/01/2006  11:07 PM                21 file.txt
                1 File(s)                21 bytes
                0 Dir(s)      2,605,862,912 bytes free

D:\test>
```

Doch auch diesmal ändert sich **nichts**.

Übrigens ... es gibt ein grafisches Werkzeug (*12Ghosts*), das beweist, daß dies nicht die ganze Wahrheit ist.

So sehen die 3 Zeitstempel aus, nachdem das *Read-Only* Bit gesetzt wurde:



Und so wird die Zugriffszeit durch die Sicherung beeinflusst - die Sekunden werden auf 0 zurückgesetzt:

